treeclimbR pinpoints the data-dependent resolution of hierarchical hypotheses

Huang, Ruizhu ; Soneson, Charlotte ; Germain, Pierre-Luc ; Schmidt, Thomas S B ; von Mering, Christian ; Robinson, Mark D

Abstract: treeclimbR is for analyzing hierarchical trees of entities, such as phylogenies or cell types, at different resolutions. It proposes multiple candidates that capture the latent signal and pinpoints branches or leaves that contain features of interest, in a data-driven way. It outperforms currently available methods on synthetic data, and we highlight the approach on various applications, including microbiome and microRNA surveys as well as single-cell cytometry and RNA-seq datasets. With the emergence of various multi-resolution genomic datasets, treeclimbR provides a thorough inspection on entities across resolutions and gives additional flexibility to uncover biological associations.

DOI: https://doi.org/10.1186/s13059-021-02368-1

Huang *et al. Genome Biology* (2021) 22:157
https://doi.org/10.1186/s13059-021-02368-1

Genome Biology

METHOD

Open Access

treeclimbR pinpoints the data-dependent resolution of hierarchical hypotheses

Ruizhu Huang[1], Charlotte Soneson[1,2], Pierre-Luc Germain[1,3], Thomas S.B. Schmidt[1,4], Christian Von Mering[1] and Mark D. Robinson[1*]

*Correspondence:
mark.robinson@imls.uzh.ch
[1]Department of Molecular Life Sciences and SIB Swiss Institute of Bioinformatics, University of Zurich, 8057 Zurich, Switzerland
Full list of author information is available at the end of the article

Abstract

treeclimbR is for analyzing hierarchical trees of entities, such as phylogenies or cell types, at different resolutions. It proposes multiple candidates that capture the latent signal and pinpoints branches or leaves that contain features of interest, in a data-driven way. It outperforms currently available methods on synthetic data, and we highlight the approach on various applications, including microbiome and microRNA surveys as well as single-cell cytometry and RNA-seq datasets. With the emergence of various multi-resolution genomic datasets, *treeclimbR* provides a thorough inspection on entities across resolutions and gives additional flexibility to uncover biological associations.

Introduction

In many fields, multiple hypotheses are simultaneously tested to investigate the association between a phenotypic outcome (e.g., disease status) and measured entities (e.g., microbial taxa). When a hierarchy of entities exists, hypotheses can be arranged in a tree structure that indicates different resolutions of interpretation. For example, in metagenomics, a tree constructed based on marker gene (or genomic) sequence provides taxonomic resolution to investigate associations between phenotype and taxa abundance. Associations tested only at a fine resolution (e.g., species-level on the taxonomic tree) might not have sufficient statistical power to detect taxa with small changes, which are of interest if they appear coherently. Given that closely related taxa often share similarity in response to environmental change [1], differential analysis performed on a broader resolution (e.g., phylum level) may improve detection by accumulating the small coherent changes. However, a broad resolution is not always desirable: using too low of a resolution cannot pinpoint specific taxa that exhibit an association. Thus, there is a need for methods that balance detection power and error control, while also giving flexibility to find the relevant resolution to interpret the data.

A similar challenge exists in the analysis of microRNA (miRNA) data. miRNAs are small non-coding RNA molecules, and their dysregulation is associated with diseases including retinal disorder, cardiovascular disease, and cancer [2, 3]. The abundance of miRNAs could be affected by regulation occuring at multiple levels of their biogenesis [4]: miRNAs in the same transcript are generally co-transcribed, but the individual miRNAs can be additionally regulated at the post-transcriptional level, and variations in Dicer cleavage or RNA editing can lead to distinct RNA fragments. A tree, where each leaf represents a unique mature miRNA sequence, and internal nodes represent miRNA duplexes, primary transcript, and clusters of miRNAs, could provide different biogenesis resolutions to interpret disease-associated miRNA dysregulation. A typical approach for microbial and miRNA surveys is so-called differential abundance (DA) analysis [5, 6], where the abundance of each entity measured is tested for association with a phenotype of interest. Regardless of the specific data, the focus becomes locating the right resolution (e.g., microbial taxa or miRNAs) that have phenotype-associated abundance changes. In such cases, the input data shares the same structure: abundance of entities collected across samples and a tree encoding the hierarchy of entities. A similar but more complicated case is so-called differential state (DS) analysis [7, 8], which arises in the analysis of single-cell datasets and typically involves comparing measurements on a single entity (e.g., cell subpopulation) across multiple samples (e.g., changes in marker intensity among markers not used in subpopulation definition). In contrast to DA tests, DS can have considerable multiplicity, with 10s to 1000s of feature profiles (e.g., antibody intensity or gene expression) for each entity. Such scenarios are usually encountered in single-cell RNA sequencing (scRNAseq) data and mass cytometry (CyTOF) data, where several reports have shown subpopulation-specific responses that occur in disease states or due to external stimuli [9–12]. Notably, the classification of cell subpopulations often requires selecting a resolution of the data, and even when well-established markers exist, a cell subpopulation might still contain hidden diversity [13, 14]. It is also unclear whether detected state changes really occur at the subpopulation level or are driven by smaller subsets of cells. In the extreme case, if changes occur at a fine resolution and in offsetting directions, they might not even be detected when the whole subpopulation is considered. It is therefore desirable to have more flexibility in the analysis, where some changes of interest occur at very specific subpopulations, while others occur among broad cell subpopulations. To achieve this, the use of a tree to store cell subpopulations on different resolutions, and exploring on the tree to find a suitable resolution, will ideally lead to better understanding of cellular response. Briefly, in the DS test, data includes a tree encoding the hierarchy of entities (cell subpopulations) and observations of multiple features (genes or antibodies) on each entity across samples. Notably, the DA test is a special case of the DS test, where each entity has only one feature: relative abundance.

Currently, several methods are available, either general for multiplicity correction or specific for a certain type of data. Yekutieli [15] proposed the hierarchical false discovery rate (*HFDR*) controlling procedure for tree-structured hypotheses. It increases power by selectively focusing on branches that are more likely to contain alternative hypotheses. Instead of generating hierarchical hypotheses, an empirical Bayes approach, *StructFDR* [16], performs hypothesis tests only on the leaf level and improves the power by incorporating a correlation matrix converted from a tree (based on distances among leaves) as the prior correlation structure to share information among hypotheses. *MiLineage* [17]

is developed for microbiome data and localizes the phenotype-associated lineages on the taxonomic tree by splitting a tree into multiple lineages, each of which includes a parent node (taxon) and its direct child nodes (taxa on a finer resolution). It then performs multivariate tests concerning multiple taxa in a lineage to test the association of lineage to a phenotypic outcome. *Phylofactor* [18] is a graph-partitioning algorithm that iteratively partitions the tree into clades to identify those having similar association pattern with the environmental metadata. *LEfSe* [19] mainly focuses on biomarker discovery of metagenomic data, by first identifying DA features using the Kruskal-Wallis sum-rank test (KW), and further selects features that have effect sizes above a specified threshold using linear discriminant analysis. In recent years, several tree-guided lasso methods have been developed. For example, *TASSO* [20] applies an $l1$ penalty on the sum of coefficients within each possible subtree, while *rare* [21] applies an $l1$ penalty on latent variables of nodes to induce subtrees having equal coefficient values. *Citrus* [22] works on CyTOF data and applies a lasso-regularized regression model [23] to automatically select stratifying subpopulations and cell response features that are the best predictors of a phenotypic outcome. An alternative to *Citrus* [22], *diffcyt* [6], over-clusters cells into subpopulations and performs differential analysis at this higher resolution separately for each feature, without any attempt to summarize concordant signal on similar cell subpopulations.

Existing methods have limitations. *HFDR* [15] does not perform well for compositional data in the DA setting because it typically stops right on the root branch, where essentially sample-level sequencing depths are compared and thus it fails to move along branches to specific entities; furthermore, no specific consideration is given to the DS case where there are multiple hypotheses (multiple features) per node: the global FDR over all features cannot be controlled at a specific level if the procedure is performed separately on each feature, and decisions of rejecting a node to move toward its child nodes cannot be taken separately for different features if the procedure is performed simultaneously on all features. *StructFDR* [16], which transforms P values into z-scores and performs z-score smoothing among leaves in close proximity (leveraging the tree structure), is powerful to identify clustered signals. However, when signals are scattered in the tree, their z-scores might be pulled down by their non-signal neighbors due to smoothing, which makes *StructFDR* less powerful than BH [24], as shown by Bichat et al. [25]; additionally, no consideration is made for the DS case where a leaf has multiple P values. *Phylofactor* requires the number of clades that the tree should be cut into, the true value of which is generally unknown in reality. *LEfSe* [19] directly applies the KW test on each feature and thus does not take confounders into consideration and might have much higher FDR than expected due to the lack of multiplicity correction. *TASSO* and *rare* are designed to regress continuous outcomes onto compositional data, which does not fully match the more general setting explored here. Lasso-regularized models [23] (e.g., *Citrus* [22]), which tend to pick one and ignore the rest among highly correlated predictors [26], can be potentially applied to pick a resolution of a relevant branch where nodes representing a cell subpopulation are nested and highly correlated. However, the automatic selection might also occur among highly correlated cell subpopulations from different branches, or features (e.g., genes) behaving similarly in the same cell subpopulations, which leads to loss of relevant information. *diffcyt* works well for the DA and DS case of CyToF data but at a fixed arbitrary resolution.

To overcome these limitations, we propose a new algorithm, *treeclimbR*, that uses the tree topology together with the molecular profiling data. We show gains in sensitivity to detect relevant entities when a tree has branches with coherent changes, and similar performance to BH [24] when the tree is uninformative. *treeclimbR* has several unique attributes: it explores the latent resolution of association by proposing multiple candidate resolutions, and it selects the optimal candidate in a data-driven way; since each candidate resolution consists of nodes that do not have ancestor-descendant relationship, *treeclimbR* can identify branches of relevant entities to show characteristics shared among them while avoiding nested nodes that are difficult to interpret. Furthermore, in DS testing, the exploration of resolution is conducted separately for each feature, which allows different features to stop at different resolutions of the tree. This matches the reality that features (e.g., gene expression) might be regulated differently in different cell subpopulations and therefore allows a more flexible data analysis platform.

Results

Overview of treeclimbR

There are many examples in the biology of entities (such as miRNAs, microbial species, or single-cell populations), measured across samples from different conditions, where it is of interest to detect associations between (the presence of) entities and condition (e.g., disease status) and where information of the relationship *between* entities could be leveraged to simplify or bring insight into the interpretation. Our new algorithm, *treeclimbR*, combines a tree that encodes the hierarchical relationship between entities with these observations, and pinpoints a suitable resolution on the tree to interpret the association. In this manuscript, branches of relevant entities that have differential abundance or expression levels among groups are called signal branches. Notably, the approach is general, in that it can be applied to interpret arbitrary statistical models that are fit at the nodes of the tree. The five main steps are illustrated in Fig. 1: (a) data aggregation, (b) differential analysis, (c) candidate proposal, (d) multiple testing correction, and (e) candidate evaluation.

The data aggregation and differential analysis generate data and statistics for internal nodes. Depending on the context, for each internal node, we either take the mean, median, or sum of the data within its descendant leaves. On each node, we compare (aggregated) data across groups to get an estimated direction of change and test a null hypothesis, H_0 (e.g., that there is no difference), resulting in a P value. As shown in Fig. 1b, the hypotheses are in a hierarchical structure that might affect the control of false discovery rate (FDR) when using methods (e.g., the Benjaminin-Hochberg procedure [24]) to correct for multiplicity. To solve the hierarchical issue, an internal node is used to represent its descendant leaves that have coherent change. As the true signal is unknown, we explore the whole tree using a search procedure that starts from the root and moves toward the leaves to capture the latent signal pattern at different resolutions, which we refer to as "candidates."

Multiple candidates are proposed, and a selection process is applied to select the optimal one. Figure 1c shows the generation of four example candidates (*C1, C2, C3, C4*) based on node-level U scores, which combine the direction and strength of the association and vary with a parameter t that has range $[0, 1]$ (see the "Methods" section). The whole tree can be scanned, and the search is stopped at different granularities (labeled as "tested"

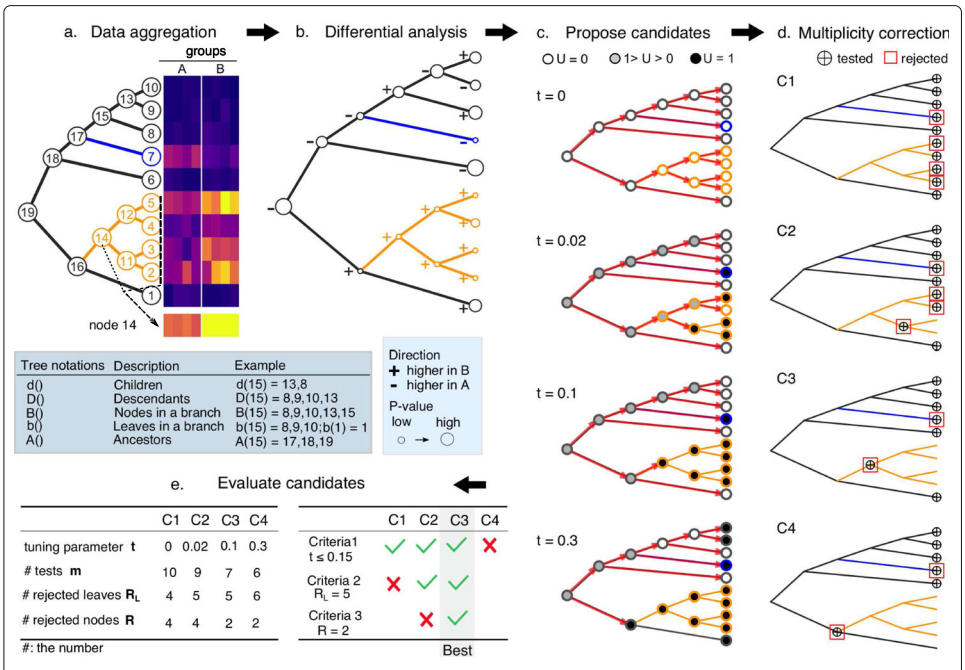

Fig. 1 Schematic overview of the *treeclimbR* algorithm. **a** Data aggregation. An example tree of entities with 10 leaves (1–10) and 9 internal nodes (11–19). Measurements of entities in samples across multiple groups (e.g., Group A, Group B) are shown in the heatmap. For internal nodes, data is generated from their descendant leaves (e.g., node 14 from nodes 2, 3, 4, and 5). Signal branches are colored in blue (higher in Group A) and orange (higher in Group B). **b** Differential analysis is run on each node to estimate its direction of change and obtain a *P* value. **c** The generation of four example candidates, *C*1, *C*2, *C*3 and *C*4. The climbing starts from the root and stops when reaching leaves or nodes with $U = 1$, a score dependent on a tuning parameter *t*. **d** Multiple testing correction on each candidate. Nodes labelled by ⊕ are termini (i.e., candidates) obtained in **c**. *C*1 is the leaf level where differential analysis is performed when a tree is not available. The null hypothesis is tested on each node of a candidate, and multiplicity is corrected within a candidate. Rejected nodes are shown in red rectangles. **e** Results from **c** are summarized in the left table. *m*, *R*, and R_L are the number of hypothesis tests (nodes with ⊕), the number of rejected nodes, and the number of rejected leaves (the descendant leaves of rejected nodes), respectively. The best candidate is selected based on three criteria. Candidates that fail in one criterion would not enter to the next. Note: tree notations used in this article are listed below **a**

in Fig. 1d) to propose multiple candidates. If the null hypothesis on an internal node is rejected, all its descendant leaves are considered to have their null hypotheses rejected. Multiple hypothesis correction is performed separately on each candidate (see Fig. 1e). The best candidate is selected by evaluating candidates according to three criteria: (i) restricting the range of *t* (to control the FDR on the leaf level), which is determined by the average size of signal branches that could be detected, and is therefore data-dependent (see the "Methods" section); (ii) selecting candidates with more rejected leaf nodes to increase the power to detect entities with signal; and (iii) selecting the signal branches with fewest internal nodes (e.g., C3 over C2 in Fig. 1d), which makes the interpretation easier and is desirable to find the right resolution.

Importantly, the procedure described in Fig. 1 is for the DA test, where each entity has one feature (i.e., relative abundance across samples). A similar overall procedure (see Additional file 1: Fig. S1) is applied to the DS case where each entity has $G (G > 1)$ features (e.g., multiple markers or genes). The only difference is in Fig. 1c, where candidates at different *t* are proposed. In order to find the candidate of a specific *t*, each of the *G* features

climb the tree ($\boldsymbol{T}$) independently. This can be imagined as a column of G trees, each of which is climbed by one of the G features. For a specified t, although G trees have the same structure ($\boldsymbol{T}$), different features might end up at different nodes as their candidates if different signal patterns exist. To perform the multiplicity correction in Fig. 1d, candidates from G features at a specific t can be pooled to form a global candidate at t (see Eq. 6). Finally, the same procedure in Fig. 1e is applied to evaluate candidates.

Performance assessment on synthetic datasets

We demonstrate the performance of *treeclimbR* against several competing methods, including *miLineage* [17], *StructFDR* [16], *HFDR* [15], BH [24], *minP* (see Additional file 1: Supplementary Note 3), *LEfSe* [19], and lasso-regularized logistic regression (*lasso*) [26] on synthetic microbial datasets (parametric and non-parametric), and two published semi-simulated single-cell mass cytometry (CyTOF) datasets [6] (*AML-sim* and *BCR-XL-sim*). Benchmark results of the method run time are in Additional file 1: Fig. S7.

Parametric synthetic microbial datasets

Operational taxonomic unit (OTU) counts are sampled from Dirichlet-multinomial distributions based on the real data in three scenarios adapted from Xiao et al. [16] (see the "Methods" section): balanced signal (*BS*), unbalanced signal (*US*), and sporadic signal (*SS*) as shown schematically in Fig. 2a and Additional file 1: Fig. S8. Instead of directly multiplying counts of selected OTUs in the treatment group by a fold change, we simulate differences by modifying parameters to introduce DA (see the "Methods" section). This ensures that relative abundance of non-DA OTUs remains fixed between the groups in the compositional data and better simulates differences of low-abundance OTUs that might otherwise have zero counts. Each scenario has two signal branches where OTUs have DA between the control and treatment groups; OTUs in the same signal branch change in the same direction. In *BS*, the fold changes of OTUs within the signal branch are fixed, whereas the fold changes in the *US* case are (in the same direction but) different in magnitude. *SS* is similar to *BS*, except that only subsets of OTUs change (the rest remain unchanged). We simulate data with signals on fixed and varying branches (see the "Methods" section and Additional file 1: Supplementary Note 1). The former has three scenarios (*BS*, *US*, and *SS*) on the same two randomly selected branches to show how methods capture different signal patterns (Fig. 2); the later varies signal branches within each scenario to compare methods under different characteristics of signal branches (Additional file 1: Fig. S9–14).

In Fig. 2, we simulate different sample sizes: 10, 25, and 50 per group for each scenario. In each combination of scenario and sample size, 100 repetitions are made. The average performance of 100 repeated simulations is shown in Fig. 2b. Both *lasso* and *miLineage* identify nested nodes and cannot pinpoint DA branches. If identified nodes that are closest to the root are used, OTUs reported by *miLineage* and *lasso* are mostly false positives (see Additional file 1: Fig. S3). Here, to minimize the FDR of *lasso* and *miLineage*, we use their identified nodes that are closest to the leaf level. Generally, methods using a tree, such as *treeclimbR*, *StructFDR* [16], and *minP*, have higher power than *BH* [24]. *HFDR* [15] is unable to detect any changes between the groups, because it starts the search from the root of the tree, which effectively represents the sequencing depth of samples, and typically stops right at the root where the null hypothesis cannot be rejected; thus, its TPR

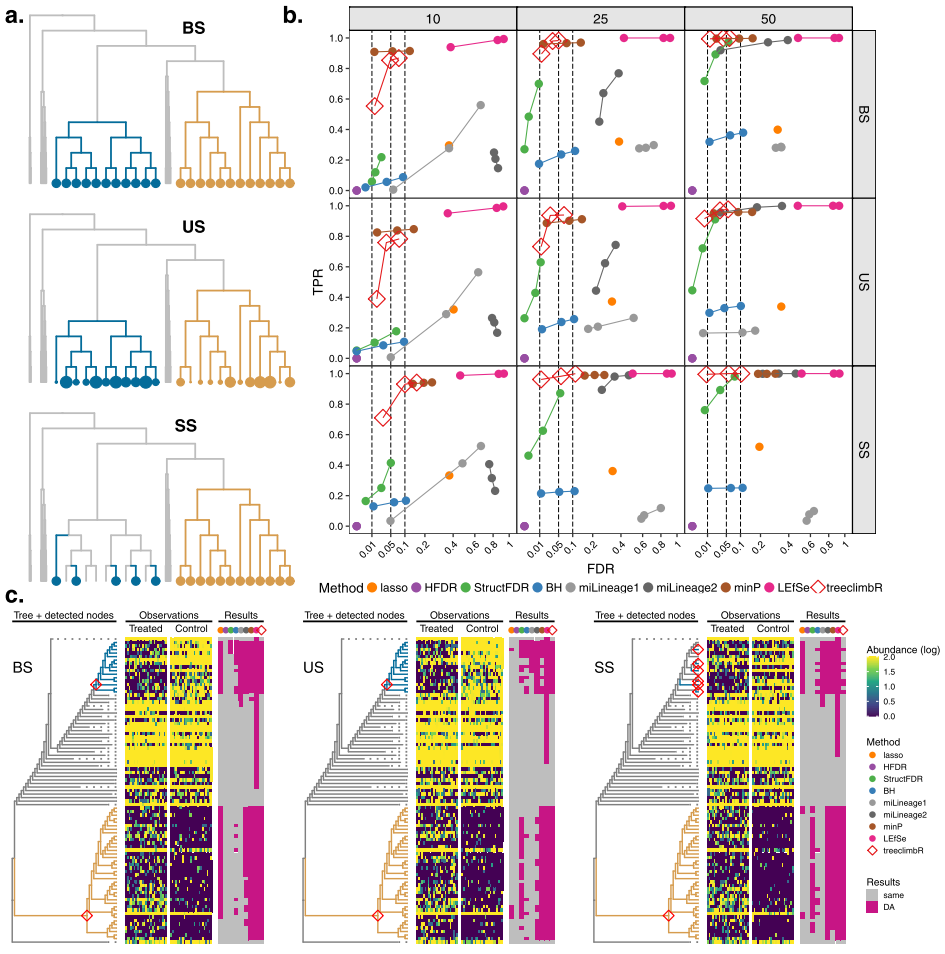

Fig. 2 The performance of methods on parametric simulated microbial data. **a** A toy tree to display a schematic example of three simulated scenarios: *BS*, *US*, and *SS*. Signal branches (i.e., with DA between groups) are in turquoise (decreased) and gold (increased), and larger points represent bigger change. **b** The average TPR and FDR (over 100 repetitions) of methods on three scenarios under different sample sizes (10, 25, and 50 per group) (see Additional file 1: Table S2 for a tabular format of the figure). Methods are in colors. Each method has three points that represent imposed FDR cutoffs at 0.01, 0.05, and 0.1. **c** DA branches identified by methods in one of the 100 repetitions. The tree has 956 leaves. The non-DA branches of the phylogenetic tree are represented with dashed lines to save space. Nodes identified by *treeclimbR* are labeled on the tree, and other methods are in Additional file 1: Fig. S2. OTU counts are shown in the heatmap with samples split by groups. Rows of the heatmap corresponding to dashed lines (consisting of many OTUs) are more gold (abundant). All OTUs (rows of heatmap) identified as DA by methods are summarized in the results panel. The results of simulations with varying signal branches are in Additional file 1: Fig. S9–14

and FDR are equal to zero. In all scenarios, *treeclimbR* outperforms others with high TPR and well-controlled FDR. *minP* performs well with high TPR in all scenarios but does not always control the FDR in the *SS* scenario where the signal does not occupy a full branch. In all three scenarios, *lasso* [26] and *miLineage* [17] have much higher FDR than expected. At a 5% FDR cutoff, OTUs identified by methods on three simulated scenarios with 25 samples per group are compared in Fig. 2c. *BH* [24] fails to find some OTUs due to low abundance or low fold change. *treeclimbR* manages to aggregate concordant signal and to stop at the right level of the tree. The two-part analysis of *miLineage* (*miLineage2*) manages to detect some OTUs with sparse counts in the gold branch, while the one-part

analysis (*miLineage1*) does not. *LEfSe* [19] identifies almost all DA branches in all simulations but with a lot of false discoveries, which may be due to the lack of multiplicity correction.

Non-parametric synthetic microbial datasets

Recently, Bichat et al. [25] have shown that currently available tree-based procedures, *StructFDR* [16] or *HFDR* [15] do not outperform the classical BH procedure when analyzing microbial data organized onto a taxonomic or phylogenetic tree. Even worse, they show that tree-based procedures might have a negative effect, giving either lower power or slightly higher power but poor FDR control compared to BH [24]. Their simulation is based on a real microbial dataset. Notably, they simulate differences by randomly selecting a set of OTUs from the most prevalent ones, and multiply counts in one of the experimental conditions by a fold change (e.g., 5). In other words, they simulate a tree that is uninformative, which provides us a negative control; we have reproduced their results and add the performance of *treeclimbR* to their non-parametric simulation (see Additional file 1: Fig. S4). Tree-based methods offer no advantage when the tree is uninformative, whereas *treeclimbR* performs in this case on par with BH [24], in terms of both power and error control.

AML-sim

We next use a dataset that simulates the phenotype of minimal residual disease in acute myeloid leukemia (AML) patients, which is designed to evaluate the performance of DA methods after clustering CyTOF profiles according to a set of lineage markers. The *AML-sim* dataset provides simulations for two subtypes of AML (cytogenetically normal (CN) and core-binding factor translocation (CBF)); only the results on subtype CN are shown. The data consists of 5 healthy and 5 synthetic "diseased" samples that are generated by spiking in a small percentage of AML blast cells from CN samples into healthy samples [6]. AML blast cells are sufficiently distinct and can typically be clustered into a separate subpopulation. Depending on the proportion of spiked-in cells, the simulated scenarios are considered as strong (5%), medium (1%), and weak (0.1%).

We follow the concept of *diffcyt* [6] to group cells into a large number of clusters using the *FlowSOM* algorithm, and compare the cell counts of clusters between the healthy and diseased groups for each cluster. Here, three different numbers of clusters have been tried: 400, 900, and 1600. A tree is built from the generated clusters based on the median expression of lineage markers (see the "Methods" section). TPR-FDR performances are shown in Fig. 3a, and a summary of each method's detections in the context of related cells is shown in Fig. 3b. *HFDR* is unable to detect the simulated signal, and has TPR and FDR both equal to 0 in all scenarios. Other methods perform well with high TPR and low FDR in the medium and the strong scenarios, and all methods fail to detect the weak signal. In the medium scenario, *diffcyt*'s TPR drops slightly when a large number of clusters (e.g., 1600) is used. For the *medium* scenario with 900 clusters, *treeclimbR*, *minP*, and *diffcyt* detect the same branch that mainly includes the AML blast cells from CN samples: *treeclimbR* and *minP* reveal an internal node, and *diffcyt* highlights the three descendant leaves of the internal node. *StructFDR* misses one leaf that contains mostly AML cells. Compared to *treeclimbR*, *lasso* identifies an additional leaf that contains mostly non-AML cells.

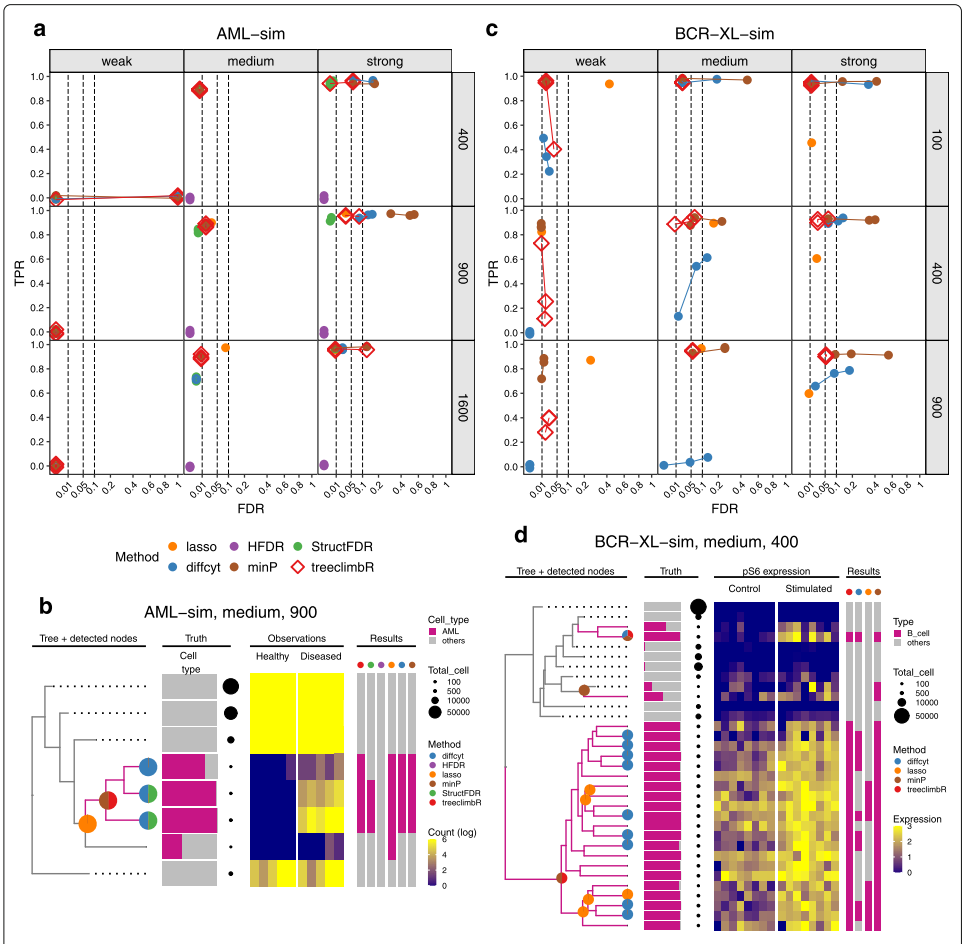

Fig. 3 Results on two semi-simulated datasets (*AML-sim* and *BCR-XL-sim*). **a** TPR vs. FDR of methods on *AML-sim* under three scenarios (weak, medium, and strong) when trees have 400, 900, and 1600 leaves (see Additional file 1: Table S3 for a tabular format of plot **a**). Methods are in colors. FDR cutoffs are at 0.01, 0.05, and 0.1. **b** DA branches identified by methods in the medium scenario of *AML-sim* using a tree with 900 leaves. The four panels show the tree, truth, observations, and results. The tree panel displays non-DA branches (without AML blast cells) in dashed lines to save space, colors branches with AML blast cells above 50% in purple, and labels nodes identified by methods; the truth panel shows cell type compositions and cell counts (point sizes) of leaves; the observation panel shows cell counts on leaves (rows) in samples (columns) split by groups; the result panel annotates leaves identified by methods. **c** same as **a**, except the *BCR-XL-sim* dataset is presented. **d** same as **b**, except the medium scenario of *BCR-XL-sim* using 400-leaf tree is presented. Here, only one feature (*pS6*) is shown

BCR-XL-sim

We next test a dataset that consists of 8 paired samples of peripheral blood mononuclear cells (PBMCs) in two treatment groups: untreated and stimulated with B cell receptor/Fc receptor cross linker (BCR-XL); the goal is to detect DS within subpopulations. Samples in the control group have healthy PBMCs, and those in stimulated group are simulated from healthy PBMCs with spiked-in B cells from BCR-XL stimulated samples [6]. In other words, samples in the two groups are different in the expression of some protein markers, including pS6, pPlcg2, pErk, and pNFkB, in B cells. The difference in marker expression profiles between the two groups is scaled to make groups distinct at three different levels: weak, medium, and strong. Cells are again grouped into a large number of clusters using *FlowSOM*, and the expression of a protein marker on each cluster is compared between

the control and the stimulated groups. Three numbers of clusters have been used: 100, 400, and 900. The tree is again built using the median expression of lineage markers in clusters.

TPR and FDR performance is calculated at the cell level, as shown in Fig. 3b. A true positive is a (spiked-in) B cell found in a DS cluster that has at least one protein marker identified as differentially expressed between the groups, and a false positive is a non-B cell found in a cluster-deemed DS. For the medium and strong scenarios, *treeclimbR* performs well with high TPR and controlled FDR; *minP* shows results similar to *treeclimbR* but with higher FDR; *diffcyt* works well with 100 clusters, but its TPR decreases as the number of clusters increases; *lasso* has slightly lower TPR than *treeclimbR* and *minP*. Signal branches identified in the medium scenario using 400 clusters are shown in Fig. 3d for a single-marker protein, *pS6*. Both *treeclimbR* and *minP* identify a large branch of B cells by picking its branch node, while *diffcyt* and *lasso* find only some of its leaves or sub-branches. In Fig. 3b, *lasso* displays almost equal TPR as *treeclimbR* because most of those missing sub-branches are identified in other marker proteins (see Additional file 1: Fig. S6). Because of the selection that *lasso* models apply, it might fail to identify some DS clusters for individual protein markers that are highly correlated with other strongly associated markers. Additionally, the result of *lasso* includes nested nodes, which can be difficult to interpret.

Tree-assisted DA and DS analyses

To highlight the diversity of applications where tree-assisted DA or DS detection arises, we applied *treeclimbR* to three datasets, including gut microbiota data, mouse miRNA data, and mouse cortex scRNAseq data.

Differential abundance of microbes in infants born differently

We applied *treeclimbR* to a public metagenomic shotgun sequencing study on fecal samples [27], with the aim to investigate whether babies born vaginally or by C-section have different microbiome compositions (see the "Methods" section). The dataset includes 464 metaOTUs from samples collected from 80 vaginally delivered infants and 15 C-section infants at different time points: 4 days (0M), 4 months (4M), and 12 months (12M), as shown in Fig. 4. Nodes reported as DA by *treeclimbR* are according to a 5% FDR cutoff. In particular, at 0M, 8 branches and 7 leaves (in total, 188 metaOTUs) are detected to be DA between C-section and vaginal babies; the difference becomes less distinct as babies grow: 2 branches and 5 leaves (65 metaOTUs) and 8 leaves are detected at 4M and 12M, respectively. The main change in composition comes from the *Bacteroides* genus, which was previously shown to be less abundant in C-section babies [28]. Vaginal babies are enriched for species in genera (e.g., *Prevotella* and *Lactobacillus*) that resemble their mother's vaginal microbiota, whereas C-section newborns tend to have higher abundance of species in genera (e.g., *Staphylococcus*) that are likely to be acquired from the hospital environment or from the mother's skin [29].

miRNA expression analysis of cardiac pressure

Similar to microbial sequences, miRNAs can be organized in a tree structure, determined not by their similarity but by their biogenesis (Fig. 5). To investigate whether miRNAs with the same origin are differentially co-expressed between mice receiving transaortic

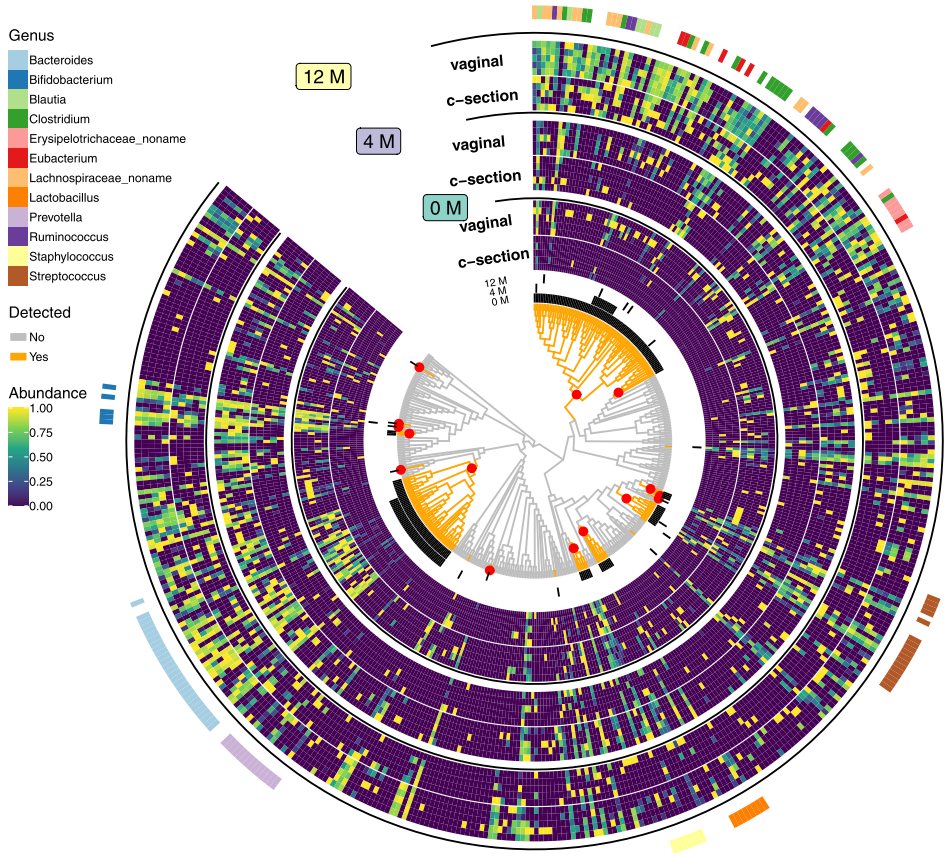

Fig. 4 The differences of the gut microbiotas between babies born vaginally and those by C-section. A phylogenetic tree of 464 metaOTUs is shown in the innermost circle. MetaOTU abundance collected from newborns at three time points: 4 days (0M), 4 months (4M), and 12 months (12M), is shown in the heatmap with log-abundance normalized to [0, 1] over samples. To avoid displaying all (285) samples, average samples that are generated by randomly assigning samples within a group into 5 categories and averaging counts of each metaOTU within each category are used. Branches detected to be DA at 0M, 4M, and 12M are shown in the circular black bars between the heatmap and the tree. On the tree, orange branches are those detected to be different at least in one time point. Red dots show nodes found by *treeclimbR* for newborns (0M). Genera that have at least 5 metaOTUs reported are shown

constriction (TAC) or mice receiving sham surgery (Sham), we ran *treeclimbR* on a subset of the dataset from Kokkonen-Simon et al. [30] (see the "Methods" section). Comparison of miRNA expression between the two groups at 5% FDR identified 166 DA nodes, representing 1250 sequences belonging to 129 miRNAs. DA nodes are identified on different levels of the hierarchy: 8 genomic clusters, 16 primary transcripts, 19 miRNAs, and 123 sequences. DA branches with at least 10 descendant leaves are annotated. Those labeled with *mixed* include miRNAs of different families, which are nonetheless transcribed from genomically clustered loci (see Additional file 1: Table S1). While many of the identified miRNAs had previously been reported in relation to cardiovascular health and disease [31–35], our analysis highlights that most of the alterations in miRNA abundance is transcriptional, including the transcriptional co-regulation of genomic clusters containing mixed miRNA families, suggesting a common reshaping of chromatin at these regions.

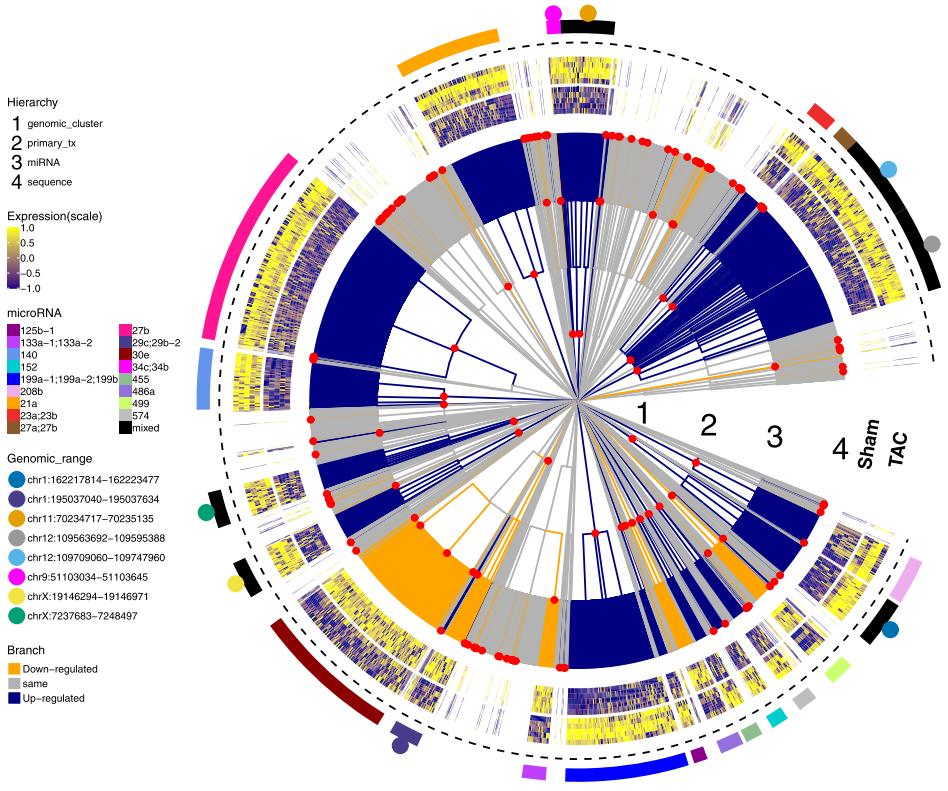

Fig. 5 Differential expression of micro-RNAs between mice treated by sham surgery (Sham) and mice treated by transverse aortic constriction (TAC). The tree organizing the miRNAs by their origin and biogenesis is shown in the innermost circle with four levels of hierarchy: level 1 includes genomic clusters (i.e., groups of miRNAs which are located in "relatively close" regions on the genome (<10 kb)); level 2 has the primary transcript or host gene; level 3 has single-strand miRNA (e.g., mmu-miR-22-3p); and level 4 has mature sequences. Nodes identified by *treeclimbR* are in red points. Up- and downregulated DA branches are shown in blue and orange, respectively. Log-expressions of sequences in identified branches are scaled and shown in the heatmap surrounding the tree. Identified branches that have at least 10 leaves are annotated, and those with more than one miRNA are labeled as mixed. The genomic ranges of identified clusters are indicated in points with different colors

DS analysis of mouse cortex scRNAseq data

To explore cell state changes (DS) on a hierarchy of cell subpopulations in scRNAseq data, we applied *treeclimbR* to understand how peripheral lipopolysaccharide (LPS) affects the brain cortex using 4 mice each from the control (vehicle) and LPS-treated groups (see the "Methods" section). The tree that encodes the hierarchical information about subpopulations, from over-clustering, comprises 66 leaves, as shown in Fig. 6a; annotation of cell types including astrocytes, endothelial cells, microglia, oligodendrocyte progenitor cells (OPC), choroid plexus ependymal (CPE) cells, oligodendrocytes, excitatory neurons, and inhibitory neurons is taken from Crowell et al. [8]. Leaves within the same cell subpopulation share similar patterns according to so-called type markers, and mostly appear in the same branch.

Using a 5% FDR threshold, *treeclimbR* identified 1561 DS genes that are expressed differently between vehicle- and LPS-treated mice in at least one cell subpopulation (genes can be deemed DS in multiple subpopulations) with absolute logFC above 1. We clustered

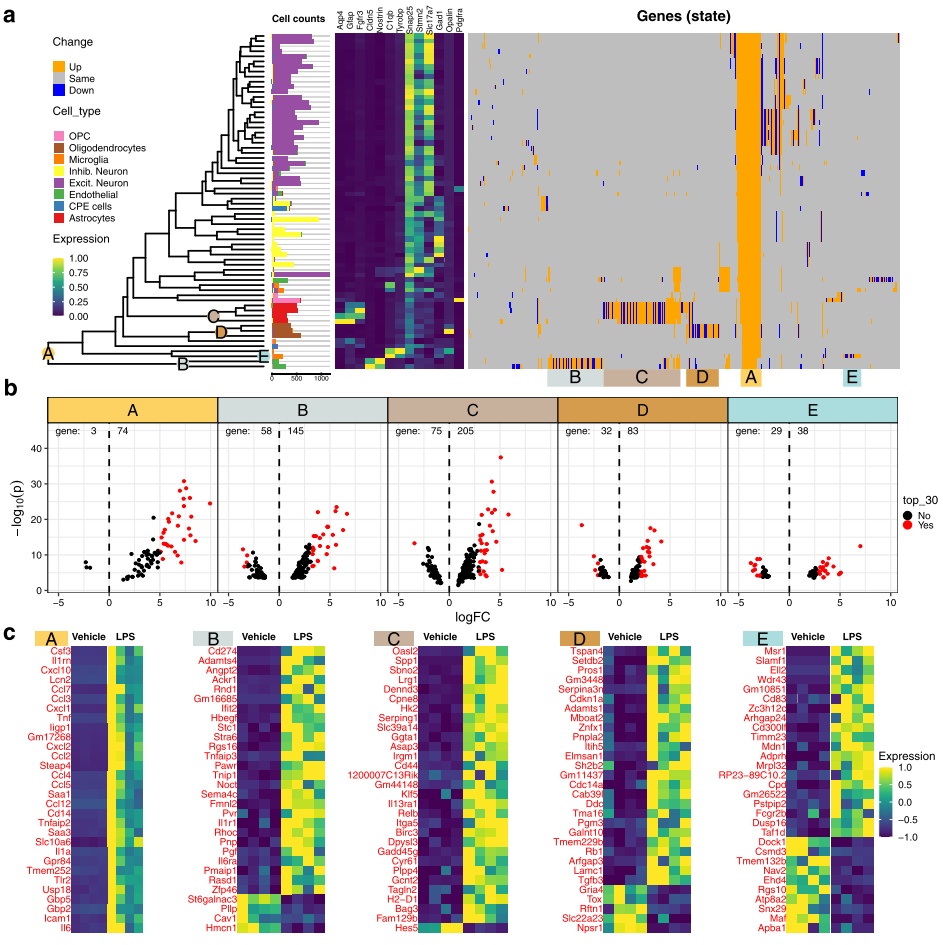

Fig. 6 Results of comparing cortex tissue from vehicle and LPS-treated mice at different resolution levels of cell types. **a** A tree encodes information of cell subpopulations at different resolutions. The cell numbers for each subpopulation are given for each leaf. The median expression of 13 canonical type markers, each columnwise scaled to [0, 1], is shown in the left heatmap. The identified DS genes are shown in the right heatmap (each column is a gene, each row subpopulation; upregulated in the LPS group in orange, downregulated in blue). **b** Volcano plots (-log *P* value versus log fold change) with the 30 genes with highest absolute logFC colored in red. Number of down- and upregulated genes are labelled on the left and right side of the dashed line at $logFC = 0$. **c** Heatmaps of top 30 genes (red points in **b**, ordered by decreasing logFC

them according to their subpopulation-wise DS pattern and summarized five distinct categories (A–E), as shown in Fig. 6. Genes in category A change across all cell types, whereas the other four categories identify genes that change in one or two specific cell types. Since *treeclimbR* is run gene-wise (one DS test for each gene at each subpopulation), different levels of the tree can be selected for each gene. To simplify the visualization, we only label nodes where more than 70% of genes in a category were selected at the shown level. For example, the level for category *B* is selected by 99.5% of genes. Volcano plots of genes in each category and sets with highest absolute log fold change are shown in Fig. 6b and c, respectively. Inflammatory signaling has been shown to trigger the upregulation of several cytokines in astrocytes [36], and indeed, we observe the upregulation of a number of them, including Cxcl2, Cxcl1, and Ccl5, not only in astrocytes but across all cell types (category A).

Discussion

Many applications in biology portray entities in a hierarchical structure. The question is then how to best leverage this information in downstream analyses where measurements (e.g., abundance) across multiple samples and experimental conditions are compared. We presented a novel principled approach, *treeclimbR*, which can be used to find a representative resolution, leading to increased power while maintaining error control. It compares favorably to leaf-level approaches (e.g., BH [24] and existing tree-based approaches (e.g., *StructFDR* [16]) when weak but coherent signals exist according to the tree.

To control FDR, *treeclimbR* assumes that leaves in branches without signal have directions up or down independently, which requires that the organization of entities on the tree is not directly driven by the changes between experimental conditions. In other words, it is recommended to have independent information on the tree and the data being analyzed. For example, in microbial data or miRNA data, the tree is organized according to sequence information (e.g., similarity or biogenesis) and the data is counts of those entities across samples. For single-cell datasets, a tree can be constructed from clustering of cell type markers, and the analysis is done on state markers, although these may not be completely independent. When the same data is used for both the tree construction and the differential analysis, we might gain power to detect relevant entities while inflating the FDR due to "double dipping" (see Additional file 1: Fig. S5). A typical example, in microbial data, is the correlation tree that is constructed based on the abundance profiles of taxa across samples from different experimental conditions. Such a tree tends to put entities showing the same direction in close proximity. In other words, it clusters not only entities with the same direction of signals in the same branch, but also those by chance appearing in the same direction. For the latter, *treeclimbR* has difficulty to distinguish it from weak but coherent signals, which overestimates the average size of signal branches r and the upper boundary of t (see Eq. 7) that would further lead to poor FDR control.

Notably, the *treeclimbR* approach is flexible, and users can specify any relevant method to perform the differential testing (DA and DS tests were the focus here, but other options are possible), and it may have applications beyond biology as long as P values and estimated directions could be provided on all nodes of the tree. To successfully obtain a representative resolution, it is important that the direction of signal is correctly estimated by the chosen method. In single-cell datasets, leaves of the tree (cell subpopulations) are usually obtained by unsupervised clustering, but the number of clusters is subjective and chosen according to a tuning parameter. Here, a balance needs to be struck between separating entities and having sufficient signal to allow methods to detect changes. In addition, users might need to preprocess the tree before running *treeclimbR*, for example, removing leaves or internal nodes that do not have sufficient data to reliably estimate directions of signals or, even separating a tree into multiple sub-trees, if entities (e.g., cell subpopulations) are sufficiently distinct.

Taken together, *treeclimbR* is a sensitive and specific method that facilitates fine-grained inferences of hierarchical hypotheses via a rooted tree. The corresponding R package is available from https://github.com/fionarhuang/treeclimbR, and the code to reproduce all analyses is available (see the "Methods" section).

Methods

Simulation framework (microbiome data)

We simulate samples for two groups: control (C) and treatment (T), and generate OTU counts ($\mathbf{x}_j^T$ or $\mathbf{x}_j^C$) in a sample j from a Dirichlet-multinomial (DM) distribution with parameters estimated from a real microbial dataset, as has been suggested in several articles [16, 17]. The real throat data, *throat_v35*, is subset from *V35* that is provided in the R package *HMP16SData* [37], by taking 153 samples collected from throat and 956 OTUs (operational taxonomic units) with non-zero count in more than 25% of samples. In particular, we sample:

$$\begin{aligned} \mathbf{x}_j^C &\sim \mathrm{DM}\left(n_j, \alpha^C\right) \\ \mathbf{x}_j^T &\sim \mathrm{DM}\left(n_j, \alpha^T\right) \end{aligned} \tag{1}$$

where $\mathbf{x}_j^C = \left(x_{1j}^C, \ldots, x_{Kj}^C\right)$ and $\mathbf{x}_j^T = \left(x_{1j}^T, \ldots, x_{Kj}^T\right)$ are counts of $K = 956$ OTUs in a sample j that belongs to control or treatment group, respectively; n_j is the total count of sample j that is randomly sampled from sequencing depths of 153 samples in *throat_v35*; $\alpha^C = \left(\alpha_1^C, \ldots, \alpha_K^C\right)$ and $\alpha^T = \left(\alpha_1^T, \ldots, \alpha_K^T\right)$ are parameters storing information about the relative abundance (proportion) and dispersion of OTUs in the control and treatment group, respectively. We estimate α^C using the R package *dirmult* [38] that reparameterizes α^C with $\pi^{\mathbf{C}} = \left(\pi_1^C, ..., \pi_K^C\right)$ and θ, where π_k^C is the expected proportion of OTU k in a sample belonging to the control group, and θ is a parameter about OTU correlation. In short, $\alpha_k^C = \pi_k^C \frac{(1-\theta)}{\theta}$. In our simulation, θ is estimated from *throat_v35* to apply in both control and treatment groups, and π^C and π^T are manipulated to create three scenarios: *BS*, *US*, and *SS* (see Fig. 2a and Additional file 1: Fig. S8). The simulated data (in the control group) is shown to have similar mean-variance relationship but a bit less random zeros when compared to the real data using *countSimQC* [39] (see Additional file 2).

In *BS*, signals are simulated on two randomly selected branches (A and B) by swapping their proportions in the treatment group as Eq. 2; *US* and *SS* are in Additional file 1: Supplementary Note 1.

$$\begin{cases} \hat{\pi}_k^T = \hat{\pi}_k^C; & k \notin A, B \\ \hat{\pi}_k^T = r\hat{\pi}_k^C; & k \in A \\ \hat{\pi}_k^T = \frac{1}{r}\hat{\pi}_k^C; & k \in B \end{cases} \tag{2}$$

where $r = \frac{\sum_{k\in B}\hat{\pi}_k^C}{\sum_{k\in A}\hat{\pi}_k^C}$ is the fold change; $\hat{\pi}_k^C$ is the estimated proportion of OTU k from *throat_v35*. In other words, π^C is estimated from *throat_v35*, and π^T is obtained based on π^C by changing values of selected OTUs.

Description of treeclimbR methodology

Data aggregation

Here, the aggregation is shown in Eqs. 3 and 4 for the DA and DS case, respectively. Depending on the dataset and method used in the differential analysis, the mean or median might be used instead of sum. In the DA case, counts of K entities in J samples are observed, and a tree on entities is constructed such that each entity can be mapped to a leaf. Data is aggregated in a way that the count of node i in sample j, Y_{ij} is generated as:

$$Y_{ij} = \sum_{k\in b(i)} Y_{kj} \quad \text{and} \quad i = 1, 2, ..., M; \quad j = 1, 2, ..., J; \quad k = 1, 2, ..., K \tag{3}$$

where $b(i)$ represents the descendant leaves of node i (see tree notations in Fig. 1); M is the total number of nodes on the tree; J is the number of samples; K is the number of entities observed.

In the DS case, we have values of G features observed on each cell from J samples, and a tree about cell subpopulations (entities) is constructed such that multiple cells are mapped to a leaf. Samples are collected from different experiment conditions. The value of feature g on node (cell subpopulation) i in sample j, Y_{ij}^g is aggregated from cells as:

$$Y_{ij}^g = \sum_{k \in (j \cap i)} Y_k^g \quad \text{and} \quad i = 1, 2, ..., M; \quad j = 1, 2, ..., J; \quad k = 1, 2, ..., K \tag{4}$$

where $k \in (j \cap i)$ means that a cell k is from sample j and belongs to subpopulation i (cell k is mapped to the descendant leaves of node i, or $k \in b(i)$); M, J, and K correspond to the total number of nodes, samples, and cells, respectively.

Differential analysis

Differential analysis is performed at all nodes of the tree. For the parametric synthetic microbial data and *AML-sim* data, we use *edgeR* to model the count data with negative binomial distribution and obtain P values via likelihood ratio tests for the following methods: *BH*, *HFDR*, *minP*, *StructFDR*, *diffcyt*, and *treeclimbR*. *miLineage* has its own way to calculate P values. For non-parametric synthetic microbial datasets, the non-parametric Wilcoxon rank sum test is used to compare the taxa's abundance between two groups, which generates P values for all benchmarked methods. For *BCR-XL-sim*, the median transformed expressions of cell state markers on each node (cell subpopulation) of the tree are compared between groups using *limma* [40], which generates P values for *diffcyt*, *minP*, and *treeclimbR*. Three real datasets (Infant gut microbiota, mouse miRNA, and mouse cortex scRNA) are all count data, and *edgeR* is used for the differential analysis.

The generation of candidates

Candidates are used to capture the latent signal pattern on the tree. The search for candidates is based on a U score defined as Eq. 5:

$$\begin{aligned} q_k(t) &= \mathbf{sign}(\theta_k)\mathbf{I}(p_k \le t) \\ U_i(t) &= \left| \frac{\sum_{k \in B(i)} q_k(t)}{n_B} \right| \end{aligned} \tag{5}$$

Here, $q_k(t)$ is a score of node k, derived from its P value p_k and estimated direction $\mathbf{sign}(\theta_k)$, under a tuning parameter t. When $p_k \le t$, $q_k(t) = 1$ with $\mathbf{sign}(\theta_k)$; otherwise, $q_k(t) = 0$. The U score of node i at t, $U_i(t)$, is the absolute average q scores over nodes in $B(i)$ that includes node i and its descendant nodes. n_B is the number of nodes in $B(i)$. The U score could be considered as a measure of coordinate change within a branch. It achieves 1 when a consistent pattern, which includes both signs in the same direction and P values below t, is observed, and it is close to 0 when nodes in a branch highly disagree on either the sign or P value. With a suitable t value, we might expect signal branches are in a consistent pattern while others that have P values following a uniform distribution $[0, 1]$ and directions arbitrary up or down on leaves are not. Since signal branches are unknown in reality, we cannot directly determine the value of t. To suggest different candidates of signal branches, the tree is explored by tuning t in the range $[0, 1]$ (see Additional file 1: Fig. S17).

A candidate at t is obtained using the procedure below:

1 It starts from the root and moves toward leaves along edges.
2 For each path, it stops when a node i having $U_i(t) = 1$ and $p_i < 0.05$ appears or the leaf is reached.

If a branch without signal by chance has the same direction, its branch node might reach $U = 1$ at high t (e.g., $t = 1$). In branches without signals, to keep candidate close to the leaf level, we hinder the selection of an internal node with a restriction $p_i < 0.05$. This means the probability of representing a three-leaf branch, without signals, using an internal node is around 0.01, and is much lower for a larger branch. P values selected in such a procedure are unbiased at different t for branches without signal and follow a uniform distribution (see Additional file 1: Fig. S16).

If multiple features exist, the procedure is carried out separately for each feature, and the global candidate at t, $\mathbf{C}(t)$, is defined as:

$$\mathbf{C}(t) = \bigcup_{g \in \mathbf{G}} C_g(t) \tag{6}$$

where $C_g(t)$ is the candidate of feature g generated at t, and G includes all features.

The selection of candidates

Correction for multiple testing is performed separately on each candidate, but FDR is controlled on the leaf level by limiting t in the range as below (see Additional file 1: Supplementary Note 2 and Fig. S15).

$$t \in [0, 2\alpha(r-1)] \tag{7}$$

where α is the nominal FDR; r is the average size of signal branches identified at FDR $= \alpha$. The branch size is the number of leaves in a branch. If $r = 1$, signals do not cluster on the tree, and the leaf level ($t = 0$) should be used. In real data, r is unknown and is estimated for a candidate $C(t)$ as:

$$\hat{r} = \frac{l}{s}$$

where s is the number of nodes with H_0 rejected on the candidate $\mathbf{C}(\mathbf{t})$, and l is the number of descendant leaves of those rejected nodes.

Candidates that are generated with $t \notin [0, 2\alpha(\hat{r}_t - 1)]$ are firstly discarded to control FDR. Those that have reported the highest number of leaves with the lowest number of nodes are then selected to increase power while keeping results as short as possible.

The preprocessing and analysis of datasets

Available methods

For *LEfSe*, the default settings of *LEfSe* that is installed with *conda* in *python 2.7* are used. For *miLineage*, we have applied both one-part (*miLineage1*) and two-part analysis (*miLineage2*) using the R package *miLineage v2.1*. For *lasso*, we build lasso-regularized logistic regression models, which consider values of features (e.g., abundance or expression) on all nodes of the tree as the explanatory variables, and the sample information (e.g, control or treatment group) as the response variable, with R package *glmnet 2.0-18* and chose model that gives the minimum mean cross-validated error. For *diffcyt* (*v1.6.0*), we use *diffcyt*'s *testDA_edgeR* and *testDS_limma* to analyze *AML-sim* and *BCR-XL-sim* datasets,

respectively. For *StructFDR* and *HFDR*, R packages *StructFDR v1.3* and *structSSI v1.1.1* are used, respectively. Inputs on nodes (e.g., *P* values) required by methods *StructFDR*, *HFDR*, *treeclimbR*, and *minP* (see Additional file 1: Supplementary Note 3) are estimated by *edgeR v3.28.0* (*treeclimbR*'s *runDA* function) in all datasets, except that *diffcyt*'s *testDS_limma* was used in *BCR-XL-sim* datasets. Unless specified, the default settings provided in R packages are used for all methods.

Parametric synthetic microbial data

To evaluate performance of methods on different signal patterns, datasets are simulated under three scenarios (*BS*, *US*, and *SS*) on two randomly selected branches using the R package *treeclimbR*'s *simData* function. More simulations with varying signal branches are provided to introduce signals on branches with different characteristics (see Additional file 1: Supplementary Note 1). Due to the swap of relative abundances between branches, the absolute logFC in *BS*, *SS*, and *US* are 1.45, 2.26, and in the range [0.02, 2.13], respectively. For each scenario, 100 repetitions that are on the same signal branches but different counts on OTUs are made. To perform DA analysis, data was aggregated using Eq. 3.

AML-sim and BCR-XL-sim

Datasets were downloaded from the *HDCytoData* [41] R package. According to cell type markers, cells were first grouped into a large number of clusters (400, 900, 1600 in *AML-sim* datasets and 100, 400, 900 in *BCR-XL-sim* datasets) using *FlowSOM* [42]. Then, among clusters, pairwise euclidean distances were computed using their median expressions of type markers to generate a dissimilarity matrix. Finally, the hierarchical clustering from *stats's hclust* [43] was applied on the matrix to create a tree on clusters.

Infant gut microbiota data

The data was downloaded from the *curatedMetagenomicData* [44] package that provides uniformly processed human microbiome data. Only samples from babies were used. This includes a count matrix with 464 metaOTUs in rows and 285 samples in columns, and a phylogenetic tree that has 464 leaves (metaOTUs) and 463 internal nodes. Samples belong to four time points: 4 days (0M), 4 months (4M), and 12 month (12M). At each time point, there are 15 samples from the C-section group and about 80 samples (80 in 0M, 81 in 4M, and 79 in 12M) from the vaginal group. Data was aggregated according to Eq. 3.

Mouse miRNA data

The data is from Kokkonen-Simon et al. [30], and 10 samples, including 5 receiving TOC and 5 receiving Sham surgery, are used. The trimming, alignment, and quantification of miRNA sequences were processed using *sports* [45], which ended up with 6375 miRNA sequences with counts in more than one sample. The tree was constructed based on the origins of the miRNA sequences: the miRNAs were grouped by primary transcript using the miRBase v22.1 annotation, and primary transcripts less than 10kb apart were further grouped into genomic clusters. It has 774 internal nodes and 6375 leaves. A leaf represents a unique sequence, and an internal node represents multiple sequences that share the same biological origin on a specific level. Data was aggregated as Eq. 3, and *edgeR* [46] was used to compare abundance between mice receiving TOC and mice receiving Sham surgery.

Mouse cortex scRNAseq data

We followed the preprocessing done by Crowell et al. [8] that annotates cells with 8 cell types. To obtain cell type markers, expressions of genes among cell types were first compared using *FindAllMarkers* (from *Seurat v3.1.1*) separately in each vehicle-treated sample to avoid selecting LPS-related state genes. For each cell type, the top 20 genes (ranked by absolute logFC) with absolute logFC above 0.5 were then selected; We further removed markers that were only identified in one sample and finally obtained 125 marker genes. Based on 135 unique marker genes (13 canonical type marker genes and 125 computationally identified marker genes), a tree that encodes information of cell subpopulations at different resolutions was constructed using *Seurat's FindClusters* (resolution at 6) and *BuildClusterTree*. The tree has 66 leaves, each of them representing a cell subpopulation. To perform DS analysis, data was aggregated as Eq. 4.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-021-02368-1.

Additional file 1: Supplementary materials
Additional file 2: countSimQC report of parametric synthetic microbial datasets
Additional file 3: Review history

Acknowledgements
The authors thank Dr. Lukas M. Weber (Johns Hopkins University) for the assistance in preprocessing two semi-simulated CyTOF data and the members of the Robinson Lab at the University of Zurich for the valuable feedback on the methodology.

Peer review information
Andrew Cosgrove was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Review history
The review history is available as Additional file 3.

Authors' contributions
RH, CS, and MDR developed *treeclimbR*. RH, CS, TSS, CVM and MDR developed the *minP* method. RH implemented the methods, the simulation framework, and the method comparison. CS assisted in several technical and conceptual aspects. RH performed the data analysis and interpretation of the CyTOF and microbial data. RH and PLG performed the data processing and analysis and interpretation of the miRNA and scRNAseq data. RH and MDR drafted the manuscript, with contributions from all authors. All authors read and approved the final manuscript.

Funding
This work was supported by the Swiss National Science Foundation (grant number 310030_175841). MDR acknowledges support from the University Research Priority Program Evolution in Action at the University of Zurich.

Availability of data and materials
Only public data were used for this study. *AML-sim* and *BCR-XL-sim* are available in R package *HDCytoData* [47]. The infant microbiome data is available in R package *curatedMetagenomicData* [44]. Mouse miRNA data [30] includes 10 samples with the following sample IDs: *GSM*3056352, *GSM*3056353, *GSM*3056354, *GSM*3056355, *GSM*3056356, *GSM*3056357, *GSM*3056358, *GSM*3056359, *GSM*3056360, and *GSM*3056361, which can be downloaded from Gene Expression Omnibus with accession ID *GSE*112056. Mouse cortex scRNAseq data [8] can be downloaded from https://doi.org/10.6084/m9.figshare.8976473.
All analyses were run in R v3.6.2 [43]. The results were visualized with *ggplot2* [48], *ggtree* [49], and our R package *TreeHeatmap* (https://github.com/fionarhuang/TreeHeatmap). Codes to reproduce results of this study are available at https://github.com/fionarhuang/treeclimbR_article. The aggregation of tree-structure data and the implementation of *treeclimbR* algorithm are based on R packages *TreeSummarizedExperiment* in Bioconductor and *treeclimbR* in GitHub: https://github.com/fionarhuang/treeclimbR, respectively. *treeclimbR* is available under the Artistic License 2.0. The version of source code used for the preparation of the manuscript is available on Zenodo [50].

Declarations

Ethics approval and consent to participate
Ethics approval is not applicable for this work.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Department of Molecular Life Sciences and SIB Swiss Institute of Bioinformatics, University of Zurich, 8057 Zurich, Switzerland. [2]Present Address: Friedrich Miescher Institute for Biomedical Research and SIB Swiss Institute of Bioinformatics, 4058 Basel, Switzerland. [3]D-HEST Institute for Neuroscience, Swiss Federal Institute of Technology, 8057 Zurich, Switzerland. [4]Present Address: European Molecular Biology Laboratory, Structural and Computational Biology Unit, 69117 Heidelberg, Germany.

Received: 29 June 2020 Accepted: 28 April 2021
Published online: 17 May 2021

References

1. Martiny JBH, Jones SE, Lennon JT, Martiny AC. Microbiomes in light of traits: a phylogenetic perspective. Science. 2015;350(6261):9323.
2. Vishnoi A, Rani S. MiRNA biogenesis and regulation of diseases: an overview. In: Methods in Molecular Biology. New York: Humana Press Inc.; 2017. p. 1–10.
3. Wang J, Liew OW, Richards AM, Chen YT. Overview of microRNAs in cardiac hypertrophy, fibrosis, and apoptosis. Int J Mol Sci. 2016;17(5):749.
4. Ha M, Narry Kim V. Regulation of microRNA biogenesis. Nat Rev Mol Cell Biol. 2014;15:509–24.
5. Lun ATL, Richard AC, Marioni JC. Testing for differential abundance in mass cytometry data. Nat Methods. 2017;14(7):707–9.
6. Weber LM, Nowicka M, Soneson C, Robinson MD. diffcyt: differential discovery in high-dimensional cytometry via high-resolution clustering. Commun Biol. 2019;2(1):1–11.
7. Nowicka M, Krieg C, Crowell HL, Weber LM, Hartmann FJ, Guglietta S, Becher B, Levesque MP, Robinson MD. CyTOF workflow: differential discovery in high-throughput high-dimensional cytometry datasets. F1000Research. 2019;6:748.
8. Crowell HL, Soneson C, Germain P-L, Calini D, Collin L, Raposo C, Malhotra D, Robinson MD. Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. Nat Commun. 2020;11(1):6077.
9. Velmeshev D, Schirmer L, Jung D, Haeussler M, Perez Y, Mayer S, Bhaduri A, Goyal N, Rowitch DH, Kriegstein AR. Single-cell genomics identifies cell type–specific molecular changes in autism. Science. 2019;364(6441):685–9.
10. Lawlor N, George J, Bolisetty M, Kursawe R, Sun L, Sivakamasundari V, Kycia I, Robson P, Stitzel ML. Single-cell transcriptomes identify human islet cell signatures and reveal cell-type-specific expression changes in type 2 diabetes. Genome Res. 2017;27(2):208–22.
11. Bhattacherjee A, Djekidel MN, Chen R, Chen W, Tuesta LM, Zhang Y. Cell type-specific transcriptional programs in mouse prefrontal cortex during adolescence and addiction. Nat Commun. 2019;10(1):1–18.
12. Mathys H, Davila-Velderrain J, Peng Z, Gao F, Mohammadi S, Young JZ, Menon M, He L, Abdurrob F, Jiang X, Martorell AJ, Ransohoff RM, Hafler BP, Bennett DA, Kellis M, Tsai LH. Single-cell transcriptomic analysis of Alzheimer's disease. Nature. 2019;570(7761):332–7.
13. Trapnell C. Defining cell types and states with single-cell genomics. Genome Res. 2015;25(10):1491–8.
14. Zeng H, Sanes JR. Neuronal cell-type classification: challenges, opportunities and the path forward. Nat Rev Neurosci. 2017;18(9):530–46.
15. Yekutieli D. Hierarchical false discovery rate–controlling methodology. J Am Stat Assoc. 2008;103(481):309–16.
16. Xiao J, Cao H, Chen J. False discovery rate control incorporating phylogenetic tree increases detection power in microbiome-wide multiple testing. Bioinformatics. 2017;33(18):2873–81.
17. Tang ZZ, Chen G, Alekseyenko AV, Li H. A general framework for association analysis of microbial communities on a taxonomic tree. Bioinformatics. 2017;33(9):1278–85.
18. Washburne AD, Silverman JD, Morton JT, Becker DJ, Crowley D, Mukherjee S, David LA, Plowright RK. Phylofactorization: a graph partitioning algorithm to identify phylogenetic scales of ecological data. Ecol Monogr. 2019;89(2):01353.
19. Segata N, Izard J, Waldron L, Gevers D, Miropolsky L, Garrett WS, Huttenhower C. Metagenomic biomarker discovery and explanation. Genome Biol. 2011;12(6):60.
20. Wang T, Zhao H. Structured subcomposition selection in regression and its application to microbiome data analysis. Ann Appl Stat. 2017;11(2):771–91.
21. Yan X, Bien J. Rare feature selection in high dimensions. J Am Stat Assoc. 2020;00(0):1–14.
22. Bruggner RV, Bodenmiller B, Dill DL, Tibshirani RJ, Nolan GP. Automated identification of stratifying signatures in cellular subpopulations. Proc Natl Acad Sci U S A. 2014;111(26):2770.
23. Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc Ser B Methodol. 1996;58(1):267–88.
24. Benjamini Y, Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. J R Stat Soc Ser B. 1995;57(1):289–300.
25. Bichat A, Plassais J, Ambroise C, Mariadassou M. Incorporating phylogenetic information in microbiome differential abundance studies has no effect on detection power and FDR control. Front Microbiol. 2020;11:649.
26. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. J Stat Softw. 2010;33(1):1–22.
27. Bäckhed F, Roswall J, Peng Y, Feng Q, Jia H, Kovatcheva-Datchary P, Li Y, Xia Y, Xie H, Zhong H, Khan MT, Zhang J, Li J, Xiao L, Al-Aama J, Zhang D, Lee YS, Kotowska D, Colding C, Tremaroli V, Yin Y, Bergman S, Xu X, Madsen L, Kristiansen K, Dahlgren J, Jun W. Dynamics and stabilization of the human gut microbiome during the first year of life. Cell Host Microbe. 2015;17(5):690–703.

28. Shao Y, Forster SC, Tsaliki E, Vervier K, Strang A, Simpson N, Kumar N, Stares MD, Rodger A, Brocklehurst P, Field N, Lawley TD. Stunted microbiota and opportunistic pathogen colonization in caesarean-section birth. Nature. 2019;574(7776):117–21.
29. Dominguez-Bello MG, Costello EK, Contreras M, Magris M, Hidalgo G, Fierer N, Knight R. Delivery mode shapes the acquisition and structure of the initial microbiota across multiple body habitats in newborns. Proc Natl Acad Sci U S A. 2010;107(26):11971–5.
30. Kokkonen-Simon KM, Saberi A, Nakamura T, Ranek MJ, Zhu G, Bedja D, Kuhn M, Halushka MK, Lee DI, Kass DA. Marked disparity of microRNA modulation by cGMP-selective PDE5 versus PDE9 inhibitors in heart disease. JCI Insight. 2018;3(15):e121739.
31. Duisters RF, Tijsen AJ, Schroen B, Leenders JJ, Lentink V, van der Made I, Herias V, van Leeuwen RE, Schellings MW, Barenbrug P, Maessen JG, Heymans S, Pinto YM, Creemers EE. miR-133 and miR-30 regulate connective tissue growth factor: implications for a role of microRNAs in myocardial matrix remodeling. Circ Res. 2009;104(2):170–8.
32. Bernardo BC, Gao XM, Winbanks CE, Boey EJH, Tham YK, Kiriazis H, Gregorevic P, Obad S, Kauppinen S, Du XJ, Lin RCY, McMullen JR. Therapeutic inhibition of the miR-34 family attenuates pathological cardiac remodeling and improves heart function. Proc Natl Acad Sci U S A. 2012;109(43):17615–20.
33. Cheng Y, Zhang C. MicroRNA-21 in cardiovascular disease. J Cardiovasc Transl Res. 2010;3(3):251–5.
34. Wang J, Song Y, Zhang Y, Xiao H, Sun Q, Hou N, Guo S, Wang Y, Fan K, Zhan D, Zha L, Cao Y, Li Z, Cheng X, Zhang Y, Yang X. Cardiomyocyte overexpression of miR-27b induces cardiac hypertrophy and dysfunction in mice. Cell Res. 2012;22(3):516–27.
35. Van Rooij E, Sutherland LB, Thatcher JE, DiMaio JM, Naseem RH, Marshall WS, Hill JA, Olson EN. Dysregulation of microRNAs after myocardial infarction reveals a role of miR-29 in cardiac fibrosis. Proc Natl Acad Sci U S A. 2008;105(35):13027–32.
36. Pedrazzi M, Patrone M, Passalacqua M, Ranzato E, Colamassaro D, Sparatore B, Pontremoli S, Melloni E. Selective proinflammatory activation of astrocytes by high-mobility group box 1 protein signaling. J Immunol. 2007;179(12):8525–32.
37. Schiffer L, Azhar R, Shepherd L, Ramos M, Geistlinger L, Huttenhower C, Dowd JB, Segata N, Waldron L. HMP16SData: efficient access to the human microbiome project through Bioconductor. Am J Epidemiol. 2019;188(6):1023–6.
38. Tvedebrink T. Overdispersion in allelic counts and θ-correction in forensic genetics. Theor Popul Biol. 2010;78(3):200–10.
39. Soneson C, Robinson MD. Towards unified quality verification of synthetic count data with countsimQC. Bioinformatics. 2018;34(4):691–2.
40. Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK. Limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Res. 2015;43(7):47.
41. Weber LM, Soneson C. HDCytoData: collection of high-dimensional cytometry benchmark datasets in Bioconductor object formats. F1000Research. 2019;8:1459.
42. Van Gassen S, Callebaut B, Van Helden MJ, Lambrecht BN, Demeester P, Dhaene T, Saeys Y. FlowSOM: using self-organizing maps for visualization and interpretation of cytometry data. Cytom Part A. 2015;87(7):636–45.
43. R Core Team. R: a language and environment for statistical computing. Vienna: R Foundation for Statistical Computing; 2019. https://www.r-project.org/.
44. Pasolli E, Schiffer L, Manghi P, Renson A, Obenchain V, Truong DT, Beghini F, Malik F, Ramos M, Dowd JB, Huttenhower C, Morgan M, Segata N, Waldron L. Accessible, curated metagenomic data through ExperimentHub. Nat Methods. 2017;14(11):1023–4.
45. Shi J, Ko EA, Sanders KM, Chen Q, Zhou T. SPORTS1. 0: a tool for annotating and profiling non-coding RNAs optimized for rRNA-and tRNA-derived small RNAs. Genomics, Proteomics Bioinforma. 2018;16(2):144–51.
46. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2009;26(1):139–40.
47. Weber LM, Soneson C. HDCytoData: collection of high-dimensional cytometry benchmark datasets in Bioconductor object formats. F1000Research. 2019;8.
48. Wickham H. ggplot2: elegant graphics for data analysis. New York: Springer; 2016.
49. Yu G, Smith DK, Zhu H, Guan Y, Lam TT. ggtree: an R package for visualization and annotation of phylogenetic trees with their covariates and other associated data. Methods Ecol Evol. 2017;8(1):28–36.
50. Huang R, Soneson C, Germain P-L, Schmidt TSB, Von Mering C, Robinson MD. treeclimbR pinpoints the data-dependent resolution of hierarchical hypotheses. Zenodo. 2021. https://doi.org/10.5281/zenodo.4679579.

Publisher's Note